<http://www.appinventor.org/book2>

http://www.appinventor.org/bookChapters/chapter1.pdf

The chapter covers the following topics:

• Building apps by selecting components and specifying their behavior.

• Using the Component Designer to select components. Some components are visible on the device’s screen and some aren’t.

• Adding media (sounds and images) to apps by uploading them from your computer.

• Using the Blocks Editor to assemble blocks that define the components’ behavior.

• Testing apps with App Inventor’s *live testing*. This lets you see how apps will look and behave on the device, step by step, even as you’re building them.

• Packaging the apps you build and downloading them to a device.

You can begin programming with App Inventor by opening a browser to ai2.appinventor.mit.edu. This opens the newest version of App Inventor, which was released in December, 2013. Some people call it App Inventor 2, but it is formally just named App Inventor, and the previous version is called App Inventor Classic.

The App Inventor programming environment has three key parts:

• **The *Component Designer***(*Figure 1-2*). You use it to select components for your app and specify their properties.

• **The *Blocks Editor***(*Figure 1-3*). You use it to specify how the components will behave (e.g., what happens when a user clicks a button).

• **An Android device** with which you can actually run and test your app as you are developing it.

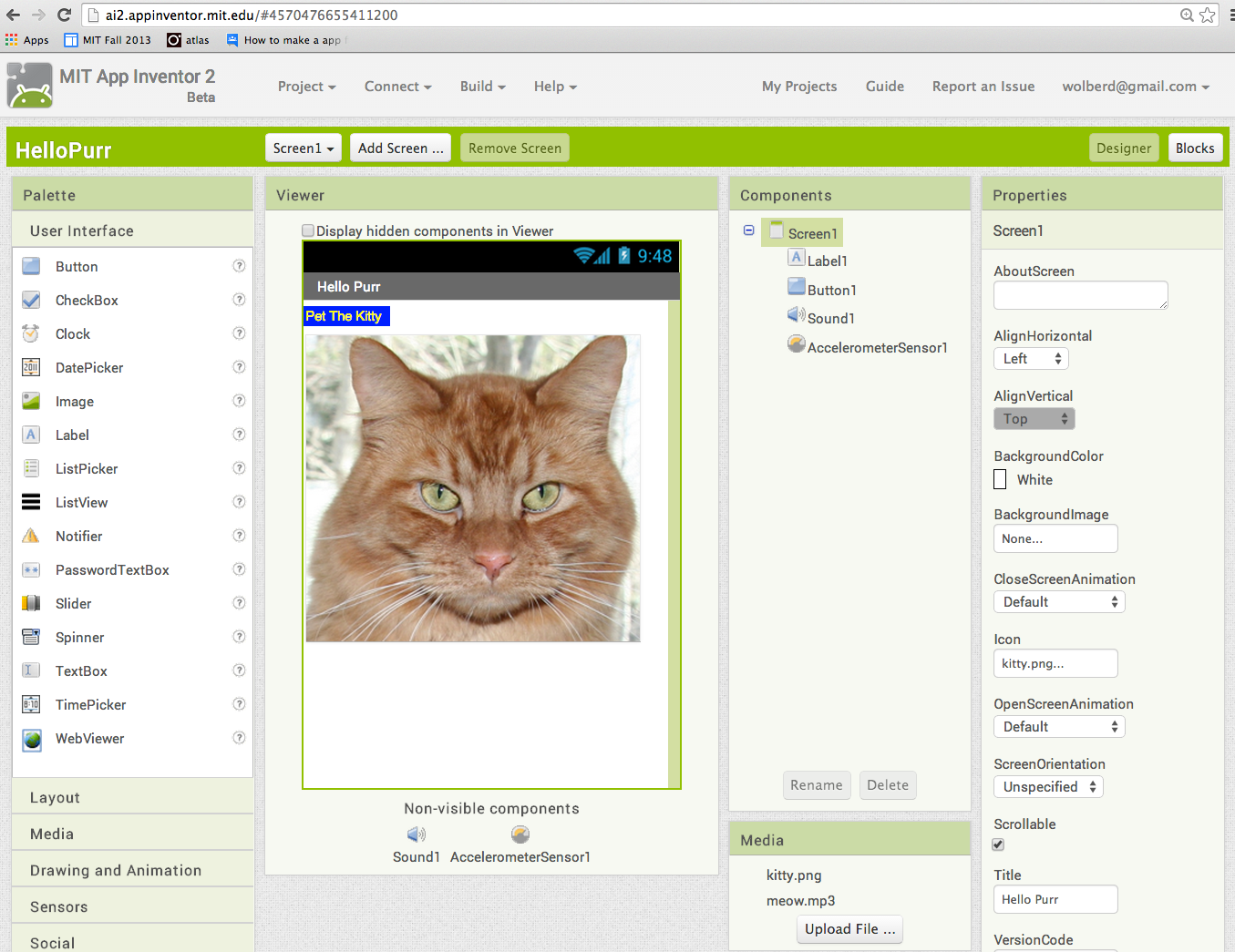


Fig. 1:2



Fig. 1:3 Block Editor

To create a project, at the upper left of the page, click “New Project,” enter the project name “HelloPurr” then click OK.

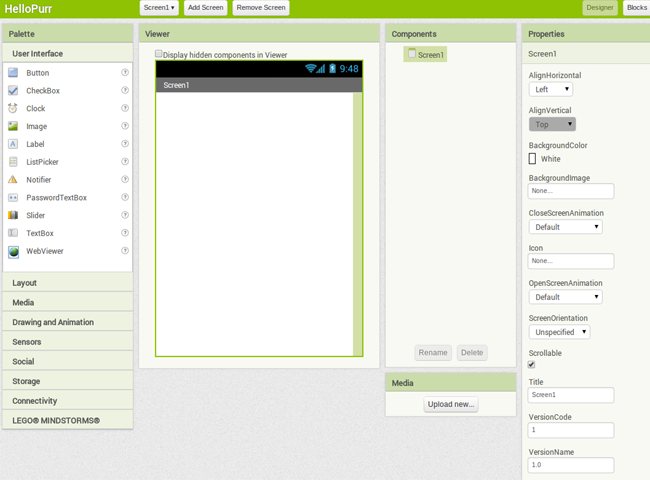
The first window that opens is the **Component Designer**. The Blocks Editor is available by clicking on the “Blocks” button in the upper-right corner of the window.

App Inventor is a cloud computing tool, meaning that your app is stored on an online server as you work. So if you close App Inventor, your app will be there when you return.

Some components are more elaborate:

* Drawing Canvas, that can hold still images or animations
* Accelerometer, a motion sensor that detects when you move or shake the device;
* components that make or send text messages, play music, and video, get information from websites, and so on

When you open the Designer, it will appear as shown in Figure 1-4.



The Designer is divided into several areas:

• Toward the center is a white area called the *Viewer*. This is where you place components

• To the left of the Viewer is the *Palette*, which is a list of components from which you can select.

• To the right of the Viewer is the *Components* list, which lists the components in your project.

• Under the Components list is an area that shows the *Media* (pictures and sound) in the project.

• To the far right is a section that shows the Properties of components; when you click a component in the Viewer, you’ll see its Properties listed here.

MAKING A LABEL

The first component to add is a Label:

1. Go to the Palette, open the User Interface drawer if it is not open, click Label (which appears about six spots down in the list of components), and drag it to the Viewer. You’ll see a rectangular shape appear on the Viewer, containing the words “Text for Label1.”

2. Look at the Properties box on the right side of the Designer. It shows the properties of the label. About halfway down, there’s a property called Text, with a box for the label’s text. Change the text to “Pet the Kitty” and press Return. You’ll see the text change in the Viewer.

3. Change the BackgroundColor of the label by clicking the box, which currently reads None, to select a color from the list that appears. Select Blue. Also change the TextColor of the label to Yellow. Finally, change the FontSize to 20. The Designer should now appear as shown in Figure 1-5. Designing the Components 5

Designing the



Table 1-1. The components you’ve added to the HelloPurr app

Button User Interface Button1 Press to make the kitty meow.

Label User Interface Label1 Shows the text “Pet the Kitty.”

Sound Media Sound1 Play the meow sound.

Add sound

1. If you haven’t downloaded the meow.mp3 file to your computer’s desktop, do so now by using this link: <http://appinventor.org/bookFiles/HelloPurr/meow.mp3>.
2. 2. Go to the Palette at the left of the Designer window and click the header marked Media to expand the Media section. Drag out a Sound component and place it in the Viewer. No matter where you drop it, it will appear in the area at the bottom of the Viewer marked “Non-visible components.” Non-visible components are objects that do things for the app but don’t appear in the visual user interface.
3. Click Sound1 to show its properties. Click the Source property and then go through the steps to upload and choose the meow.mp3 file you downloaded earlier. When you’re done, you should see both kitty.png and meow.mp3 listed in the Media section of the Designer.

Live Testing

If you have an Android device and an internet connection with WiFi, you can set up live testing in minutes, and you don’t have to download any software to your computer (just an app on your phone).

If you don’t have an Android device, you’ll need to perform some additional setup in order to use the emulator, the details of which are covered at <http://appinventor.mit.edu/explore/ai2/setup.html>.

If you have an Android device, do the following:

1. On your device, download and install the “MIT AI2 Companion” app from the Google Play Store. Launch the app when it’s installed.
2. Connect both your computer and your device to the same WiFi connection.
3. In App Inventor (in the browser), from the top menu, select Connect and then choose AI Companion, as shown in Figure 1-7.



Figure 1-7. Click Connect and then select AI Companion

1. On your device, launch the app you installed, the MIT AI2 Companion, as shown in Figure 1-8. Select “Scan QR Code” and then hold your device up to the QR code on the computer screen to scan it.



Figure 1-8. On your device, open the Companion app and click “Scan QR Code”

If all goes well, you should see the HelloPurr app running on your device, including all of the components you added. As you make changes in the App Inventor Designer or Blocks Editor, those changes will also appear on the device, as well.

Live testing setup If you have trouble setting up live testing, visit [*http://appinventor.mit.edu/explore/ai2/setup.html*](http://appinventor.mit.edu/explore/ai2/setup.html).

If your app does appear on the device, go ahead and tap the button. Do you think anything will happen? It won’t, because you haven’t instructed the button to do anything yet

ADDID BEHAVIOR

MAKING THE KITTY MEOW

At the top left of the window, beneath the Blocks header, you’ll see a column that includes a Built-in drawer and a drawer for each component you created in the Designer: Button1, Label1, Screen1, and Sound1. When you click a drawer, you get a bunch of options (blocks) for that component. Click the drawer for Button1. The drawer opens, showing a selection of blocks that you can use to build the button’s behavior, starting with Button1.Click at the top, as shown in Figure 1-9.

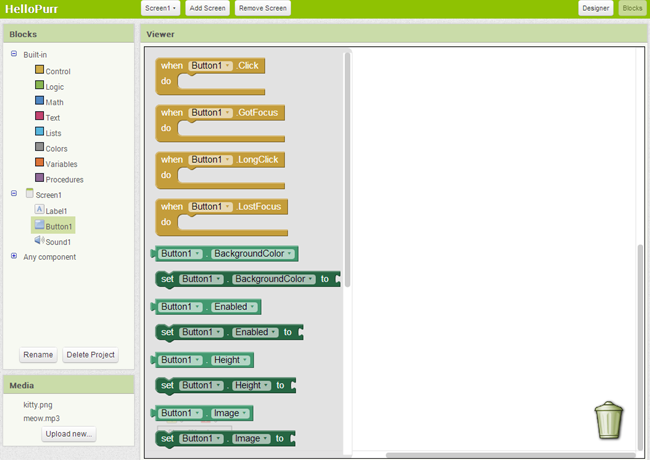
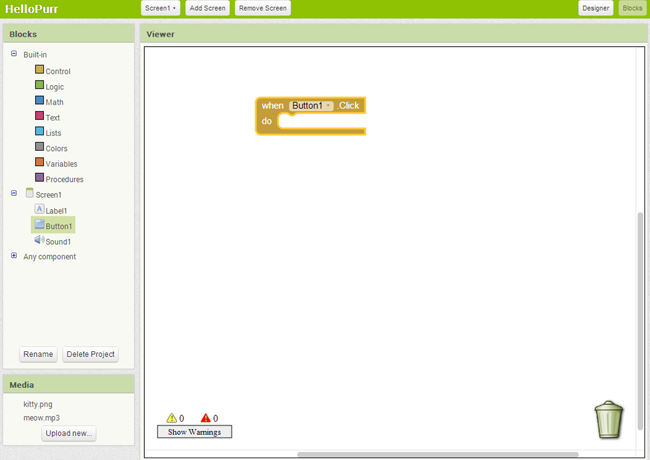


Figure 1-9. Clicking Button1 shows the component’s blocks

Click the block labeled Button1.Click and drag it into the workspace. You’ll notice that the word “**when**” is included on the **Button1.Click block**. Blocks including the word “when” are called event handlers; they specify what components should do when some particular event happens.

In this case, the event we’re interested in happens when the app user taps the image of the kitty (which is really a button), as shown in Figure 1-10. Next, you’ll add some blocks to program what will happen in response to that event.



Click Sound1 to open the drawer for the sound component, and then drag out the call Sound1.Play block. (Remember, earlier we set the property for Sound1 to the meow sound file you downloaded to your computer.)

At this point, you might have noticed that the call Sound1.Play block is shaped so that it can fit into a gap marked “do” in the Button1.Click block. App Inventor is set up so that only certain blocks fit together; this way, you always know you’re connecting blocks that actually work together. In this case, blocks with the word “call” cause components to do things.

